NAT'L INST. OF STAND & TECH R.I.C.
A11103 712533

NIST
PUBLICATIONS

Applied and
Computational
Mathematics
Division

NISTIR 4702

Computing and Applied Mathematics Laboratory

A Merit Function for Inequality Constrained Nonlinear Programming Problems

P.T. Boggs, J.W. Tolle, A.J. Kearsley

October 1991

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
Gaithersburg, MD 20899

QC
100
.U56
#4702
1991
C.2

NIST
QC100
.U56
4702
1991
c.2

NISTIR 4702

A Merit Function for Inequality Constrained Nonlinear Programming Problems

P. T. Boggs
J. W. Tolle
A. J. Kearsley

U.S. DEPARTMENT OF COMMERCE
National Institute of Standards and Technology
Computing and Applied Mathematics Laboratory
Applied and Computational Mathematics Division
Gaithersburg, MD 20899

October 1991

U.S. DEPARTMENT OF COMMERCE
Robert A. Mosbacher, Secretary
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
John W. Lyons, Director

A Merit Function for Inequality Constrained Nonlinear Programming Problems *

Paul T. Boggs † Jon W. Tolle ‡ Anthony J. Kearsley §

September 20, 1991

Abstract

We consider the use of the sequential quadratic programming (SQP) technique for solving the inequality constrained minimization problem $\min_x f(x)$ subject to: $g_i(x) \leq 0,\ i = 1, \ldots, m$. SQP methods require the use of an auxiliary function, called a merit function or line-search function, for assessing the steps that are generated. We derive a merit function by adding slack variables to create an equality constrained problem and then using the merit function developed earlier by the authors for the equality constrained case. We stress that we *do not* solve the slack variable problem, but only use it to construct the merit function. The resulting function is simplified in a certain way that leads to an effective procedure for updating the squares of the slack variables. The final form of the merit function has many desirable properties, and is demonstrated to be effective in practice.

---

*Contribution of the National Institute of Standards and Technology and not subject to copyright in the United States. This research partially supported by AFOSR Contract #88-0267

†Applied and Computational Mathematics Division, National Institute of Standards and Technology, Gaithersburg, MD 20899

‡Mathematics Department, University of North Carolina, Chapel Hill, NC 27599

§Department of Mathematical Sciences, Rice University, Houston, Texas 77251

1. Introduction

We consider the inequality-constrained minimization problem,

$$\begin{array}{l} \min_x f(x) \\ \text{subject to: } g(x) \leq 0 \end{array} \tag{1.1}$$

where $x \in \Re^n$, and $f : \Re^n \rightarrow \Re$, and $g : \Re^n \rightarrow \Re^m$ are smooth functions. One of the most successful methods for solving (1.1) is the sequential quadratic programming (SQP) method in which at each iteration a quadratic program is solved to obtain the step direction toward the next iterate. In particular, given a current approximation, x^c, to the solution, x^*, one forms the quadratic program

$$\begin{array}{l} \min_\delta \nabla f(x^c)^\mathsf{T}\delta + \frac{1}{2}\delta^\mathsf{T} B^c \delta \\ \text{subject to: } \nabla g(x^c)^\mathsf{T}\delta + g(x^c) \leq 0 \end{array} \tag{1.2}$$

where B^c is usually taken to be a current approximation to the Hessian of the Lagrangian of (1.1). Let δ^c be the solution of (1.2). Then the next approximation, x^+, is calculated by

$$x^+ = x^c + \alpha\delta^c \tag{1.3}$$

where α is a scalar *steplength*. This procedure is repeated until convergence.

Actual algorithms based on the above have been formulated and coded, and have performed well in practice on a wide variety of problems. Certain aspects of the general SQP scheme are still the subject of much current research, and the theory that explains the performance is still incomplete. In this paper we address one issue that has not yet been satisfactorily resolved; namely, a means of chooosing an appropriate steplength, α, so as to guarantee global convergence. In particular, we propose a *merit function* for the inequality-constrained problem (1.1). The approach we take is based on our earlier work, [BogT84] and [BogT89], in which a merit function was provided for equality-constrained problems.

By a merit function we mean a function that can be used as a test to assure that the potential step given by (1.3) will make x^+ a better approximation to the solution than x^c. For unconstrained optimization problems, the objective function serves this purpose; one simply chooses α such that

$$f(x^+) < f(x^c)$$

(with appropriate restrictions on α to assure a sufficient decrease is achieved). For constrained optimization, the possible decrease in the objective value must be weighed against the requirement that feasibility must

also be achieved — at least in the limit. Therefore, if one is using a method, such as an SQP algorithm, that does not maintain feasibility at each step, the objective function by itself is not an acceptable merit function.

Numerous papers have been published that set forth suggested merit functions; the reader is referred to [BogT89] for a survey of some of the more recent research. Most of the work that has appeared has been directed towards the equality-constrained problem; straightforward generalizations to the inequality-constrained problem are made difficult by the complicated nature and, in many cases, the nondifferentiablility of the merit function on the boundary of the feasible region. Work that has been directed to, or encompasses, the inequality-constrained problem includes that of [Han77], [ChaLLP82], [Sch83], and [GilMS86]. Han, in his initial papers on the SQP methods employed an l_1 penalty function for his merit function. Chamberlain et al. developed their "watchdog" technique to correct a deficiency in Han's merit function that permitted a step length of one to be rejected near the solution, thereby possibly precluding superlinear convergence. Schittkowski used an augmented Lagrangian penalty function that seems to work well on highly nonlinear problems but does not perform so well on problems where the active constraint gradients at optimality are nearly linearly dependent. Gill *et al.* have incorporated a version of Schittowski's merit function into their NPSOL program.

The merit function proposed here differs from most, if not all, others in that it does not have an unconstrained local minimum at x^* but, rather, a constrained minimum at (x^*, z^*) where the variables z are *nonnegatively constrained* slack variables. The implementation of the SQP algorithm proposed for this merit function will assure that these slack variables never become negative so that, practically speaking, the constraints are of no consequence in the computation. This merit function has excellent theoretical properties; it is smooth in its variables and exact. Moreover, locally the SQP step will always be a descent direction for this function and the length of the step will not be so restricted as to prevent superlinear convergence, i.e., the Maratos effect [ChaLPP82] will not occur. As in [BogT89] the algorithm given herein will use the merit function in a modified form that makes it computationally attractive without sacrificing its central theoretical properties and, in addition, yields global convergence.

An outline of the paper is as follows. In §2 we motivate our proposed merit function and establish its basic properties. In §3 we suggest a modification of the function to overcome certain deficiencies, and we incorporate these ideas in a somewhat detailed algorithm. §3 also contains the results of numerical experiments and comments on their significance.

Because the structure and spirit of this paper is similar to that of [BogT89] we neither give complete proofs of every proposition nor do we give detailed statements of convergence theorems. These omissions can be filled in; but only at some cost in simplicity of exposition and with no compensating increase in clarity and insight. Where these omissions occur the reader is directed to our earlier work.

2. A Merit Function for the Inequality Constrained Problem

In [BogT84] the authors introduced a merit function for use in solving equality-constrained nonlinear programs by an SQP method. While possessing several desirable properties this merit function suffered, in practice, from two major deficiencies: it did not yield global convergence and it was expensive to evaluate. A procedure employing a modified version of this merit function that overcame these difficulties was introduced in [BogT89]. This strategy was shown to be competitive in [ShaP89].

In this section we present a merit function for use with an SQP method for solving inequality-constrained problems. This new merit function is motivated by introducing squared slack variables and applying our earlier merit function to the resulting equality-constrained problem. It should be stressed, however, that the result is not simply an extension of our earlier work to the problem with these squared slack variables; there are several significant changes that are required. First, we do not solve the slack variable equality-constrained problem; we solve (1.1) by solving the approximating inequality-constrained quadratic subproblems (1.2). Second, our merit function does not update the slack variables themselves but, rather, their squares. Finally, the solution to the original problem corresponds to a *constrained* minimum of the merit function. A version of this approach was first outlined in [BogT88]; here a revised and more complete development is given.

As suggested, we motivate our merit function by first converting (1.1) to an equality-constrained problem by the addition of squared slack variables, yielding

$$\begin{aligned} &\min_{x,s} f(x) \\ &\text{subject to: } g(x) + \tfrac{1}{4}S^2 e = 0 \end{aligned} \tag{2.1}$$

where

$$S = \operatorname{diag}\{s_1, \ldots, s_m\}.$$

Here, as elsewhere in this paper, e represents a vector of ones. (The constant $\frac{1}{4}$ is chosen for convenience.)

It is well-known that if x^* is a solution to (1.1) then (x^*, s^*) with s_i^* defined to be $(-g_i(x^*))^{1/2}$ is a solution to (2.1) and conversely. Most researchers consider the actual solution of (2.1) rather than (1.1) to be impractical, for both computational and theoretical reasons. However, following [BogT84], we can construct a merit function for (2.1) of the form

$$\phi_d(x,s) = f(x) + \lambda(x,s)^{\mathsf{T}} c(x,s) + \frac{1}{d} c(x,s)^{\mathsf{T}} A(x,s)^{-1} c(x,s)$$

where

$$c(x,s) = g(x) + \frac{1}{4} S^2 e,$$

$$\begin{aligned} A(x,s) &= \nabla c(x,s)^{\mathsf{T}} \nabla c(x,s) \\ &= \nabla g(x)^{\mathsf{T}} \nabla g(x) + \frac{1}{4} S^2, \end{aligned}$$

and

$$\begin{aligned} \lambda(x,s) &= -A(x,s)^{-1} \nabla c(x,s) \begin{pmatrix} \nabla f(x) \\ 0 \end{pmatrix} \\ &= -A(x,s)^{-1} \nabla g(x)^{\mathsf{T}} \nabla f(x). \end{aligned}$$

The small scalar parameter d is chosen to penalize infeasibility. Observe that $\lambda(x,s)$ is the least squares estimate of the multiplier vector that solves the first order optimality conditions for (2.1):

$$\nabla \left[f(x) + \lambda^{\mathsf{T}} c(x,s) \right] = 0.$$

Here, and in the remainder of the paper, the symbol $\nabla h(u,v)$ refers to the differentiation of the function h with respect to all of its variables. Differentiation with respect to a subset of the variables is denoted by a subscript, e.g., $\nabla_u h(u,v)$ means the vector of partial derivatives of h with respect to the variables in the vector u. A similar convention holds for second order derivatives.

The observation that $\phi_d(x,s)$ depends only on the squared components of s leads us to replace the term $\frac{1}{4} s_i^2$ by the vector z_i and to consider the resulting function:

$$\psi_d(x,z) = f(x) + \bar{\lambda}(x,z)^{\mathsf{T}} \bar{c}(x,z) + \frac{1}{d} \bar{c}(x,z) \bar{A}(x,z)^{-1} \bar{c}(x,z)$$

where

$$\begin{aligned} \bar{c}(x,z) &= g(x) + Ze \\ \bar{A}(x,z) &= \nabla g(x)^{\mathsf{T}} \nabla g(x) + Z \\ \bar{\lambda}(x,z) &= -\bar{A}(x,z)^{-1} \nabla g(x)^{\mathsf{T}} \nabla f(x) \end{aligned}$$

and

$$Z = \operatorname{diag}\{z_1, \ldots, z_m\}.$$

It is important to note that $\psi_d(x,z)$ is *not* the function that would result from constructing the merit function for (2.1) after replacing $\frac{1}{4}s_i^2$ by z_i . This observation notwithstanding we consider the z_i as nonnegative slack variables for (1.1), and, as a consequence, we have the constrained minimization of ψ_d:

$$\begin{aligned} &\min_{x,z} \psi_d(x,z) \\ &\text{subject to: } z \geq 0. \end{aligned} \tag{2.2}$$

Our strategy for using the merit function always maintains $z \geq 0$ so that the nonnegativity constraints create neither theoretical nor computational difficulties.

In the following we develop the theoretical properties of the merit function $\psi_d(x,z)$ and relate the solutions of the problems (1.1) and (2.2). We use the term "strong local solution" to mean an optimal point, together with a multiplier vector, of a constrained optimization problem at which the following hold.

A1: The active constraint gradients are linearly independent.
A2: Strict complementary slackness holds.
A3: The second order sufficient conditions hold.

It is well known that such a strong solution is isolated and is stable with respect to small perturbations in the data of the problem. It can also be inferred, following the lines of argument in [Rob74] that if $\{x^k\}$ is a sequence converging to x^*, a strong local solution to (1.1), and $\{B^k\}$ is a sequence of uniformly bounded positive definite matrices, then the active constraints of (1.1) at x^* and the active constraints of (1.2) at x^k will coincide for k sufficiently large. This is an essential result for the theoretical justification of our proposed algorithm, as well as for most other SQP algorithms. In particular it means that the necessary and sufficient conditions for superlinear convergence of an SQP method for equality-constrained problems given in [BogTW82] can be carried over to the inequality-constrained problem in

a straightforward manner by using the projection onto the null space of the active constraint gradients.

In order to employ the merit function $\psi_d(x, z)$ we must be sure that it is well-defined, i.e., that the matrix $\bar{A}(x, z)$ is nonsingular. To formulate this condition in a convenient manner we partition the index set of the constraints into two subsets a and u. Accordingly, we can write, without loss of generality,

$$g(x) = \begin{pmatrix} g_a(x) \\ g_u(x) \end{pmatrix}$$

and, in a corresponding manner,

$$z = \begin{pmatrix} z_a \\ z_u \end{pmatrix}.$$

Usually the index subset a corresponds to the set of active constraints for either (1.1) or (1.2). For example, we could set $a = \{i : g_i(x) = 0\}$ and hence $u = \{i : g_i(x) \neq 0\}$. We now introduce the additional assumption in terms of a particular partition.

A4: The set $\{\nabla g_i(x) : i \in a\}$ is linearly independent and $z_u > 0$.

With this assumption the following proposition can be proven.

Proposition 1. *Assume that either* **A4** *holds at* (x, z) *for some partition* (a, u) *or* $z > 0$. *Then* $\psi_d(x, z)$ *is well-defined in a neighborhood of* (x, z).

If x^* is a strong solution for (1.1) and $z^* = -g(x^*)$, then **A4** holds at (x^*, z^*) and by Proposition 1 $\psi_d(x, z)$ is well-defined in a neighborhood of (x^*, z^*). We can now show that the solutions to (1.1) lead to special solutions of (2.2) and vice versa.

Proposition 2. *Let* x^* *be a strong local solution to (1.1) having multiplier vector* λ^*. *Then if* $z^* = -g(x^*)$ *and* d *is sufficiently small it is the case that* (x^*, z^*) *is a strong local solution to (2.2) with multiplier* $\bar{\lambda}(x^*, z^*) = \lambda^*$.

Proof: We note that the conditions on x^* and z^* assure that **A4** is satisfied at (x^*, z^*) and hence that $\psi_d(x, z)$ is well-defined in a neighborhood of (x^*, z^*). Also $\bar{c}(x^*, z^*) = 0$. Thus

$$\nabla_x \psi_d(x^*, z^*) = \nabla f(x^*) + \nabla g(x^*) \bar{\lambda}(x^*, z^*)$$

and

$$\nabla_z \psi_d(x^*, z^*) = \bar{\lambda}(x^*, z^*).$$

From its definition, $\bar{\lambda}(x, z)$ is seen to be the least squares solution of

$$\begin{pmatrix} \nabla g(x) \\ Z^{1/2} \end{pmatrix} \bar{\lambda} + \begin{pmatrix} \nabla f(x) \\ 0 \end{pmatrix} = 0. \tag{2.3}$$

But from the properties of x^* and the definition of z^* it follows that $z_i^* \lambda_i^* = 0$ for each i. Thus λ^* is an exact solution of (2.3) at (x^*, z^*), and so $\bar{\lambda}(x^*, z^*)$ must be λ^*. It now follows that $\nabla_x \psi_d(x^*, z^*) = 0$, $\nabla_z \psi_d(x^*, z^*) = \lambda^*$, and strict complementary slackness holds. Therefore (x^*, z^*, λ^*) is a solution to (2.2). The proof that **A3** holds for this problem is straightforward.

□

The proof of the converse follows essentially the same argument.

Proposition 3. *Let (x^*, z^*) be a strong optimal solution to (2.2) having multiplier λ^* and satisfying $g(x^*) + z^* = 0$ and $z^* \geq 0$. Then x^* is a strong optimal solution to (1.1) with multiplier λ^* and $\lambda^* = \bar{\lambda}(x^*, z^*)$.*

The fundamental part of the algorithm described in the next section determines the direction of the step for changes in the current iterates x and z. Given the current values x, z, and positive definite Hessian approximation B, the step direction δ is obtained as the solution of (1.2) and the step direction for z is taken to be q where

$$q = -\left[\nabla g(x)^{\mathsf{T}} \delta + g(x) + z\right]. \tag{2.4}$$

Thus the new iterates will be $(x + \alpha\delta, z + \alpha q)$ for some $\alpha \geq 0$. It is easily seen from the constraints of (1.2) that if the initial value of z is nonnegative then all succeeding values of z will also be nonnegative provided that α is not greater than one.

The choice of q given in (2.4) can be motivated by considering the slack variable problem

$$\begin{aligned} &\min_{x,z} f(x) \\ &\text{subject to:} \quad \begin{aligned} g(x) + z &= 0 \\ z &\geq 0. \end{aligned} \end{aligned}$$

The structured quadratic programming approximation to this problem can be seen, after some algebra, to have the form

$$\begin{aligned} &\min_{\delta, q} \nabla f(x)^{\mathsf{T}} \delta + \tfrac{1}{2} \delta^{\mathsf{T}} B \delta \\ &\text{subject to:} \quad \begin{aligned} \nabla g(x)^{\mathsf{T}} \delta + q + g(x) + z &= 0 \\ q + z &\geq 0. \end{aligned} \end{aligned}$$

The equality constraint gives the relation (2.4) while the fact that the linearized constraint in (1.2) is always satisfied implies that $q + z$ will always be nonnegative.

Before going further we give some formulas that will be useful in establishing later results. We assume in the following that $\bar{A}(x, z)$ is nonsingular.

Since

$$\nabla \bar{c}(x, z) = \begin{pmatrix} \nabla g(x) \\ I \end{pmatrix}$$

and, from (2.4),

$$\bar{c}(x, z) = g(x) + z = -\Big(\nabla g(x)^{\mathsf{T}} \delta + q\Big), \tag{2.5}$$

we have

$$\nabla \bar{c}(x, z)^{\mathsf{T}} \begin{pmatrix} \delta \\ q \end{pmatrix} = \Big(\nabla g(x)^{\mathsf{T}} \delta + q\Big) = -\bar{c}(x, z). \tag{2.6}$$

Two useful formulas that can be derived from the expression for $\bar{\lambda}(x, z)$ are

$$\nabla_x \bar{\lambda}(x, z) = -W(x, \nabla_x L(x, \bar{\lambda}(x, z))) - \nabla_{xx} L(x, \bar{\lambda}(x, z)) \nabla g(x) \bar{A}(x, z)^{-1} \tag{2.7}$$

and

$$\nabla_z \bar{\lambda}(x, z) = -\Lambda \bar{A}(x, z)^{-1} \tag{2.8}$$

where $L(x, v) = f(x) + g(x)^{\mathsf{T}} v$ is the Lagrangian function, $W(x, y) = O(y)$ uniformly in x, and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_m)$. Also we note that if λ is the multiplier for (1.2) then the KKT condition for (1.2) is

$$\nabla_x L(x, \lambda) = \nabla f(x) + \nabla g(x) \lambda = -B\delta. \tag{2.9}$$

This formula together with the complementary slackness conditions for (1.2) and the definitions of $\bar{A}(x, z)$ and q imply

$$\bar{\lambda}(x, z) - \lambda = \bar{A}(x, z)^{-1} \nabla g(x)^{\mathsf{T}} B\delta - \bar{A}(x, z)^{-1} \Lambda q. \tag{2.10}$$

We now identify some of the properties enjoyed by the merit function relative to the step direction (δ, q). An additional assumption that will hold throughout the remainder of this paper is required.

A5: The matrices B appearing in (1.2) are uniformly positive definite.

First we observe that the step potentially decreases the (x, z)-infeasibility as measured by

$$r(x, z) = |\bar{c}(x, z)|^2.$$

Indeed, as a result of (2.6) we have

Proposition 4. *Let* **A1** *and* **A4** *hold at* (x, z) *and let* δ *and* q *be defined as above. Then*

$$\nabla r(x,z)^{\mathsf{T}} \begin{pmatrix} \delta \\ q \end{pmatrix} = -2r(x,z).$$

Thus, unless $\bar{c}(x,z) = 0$ the step is always a descent step for $r(x,z)$. The next proposition shows that when $\bar{c}(x,z)$ is close enough to zero the direction is also one of descent for $\psi_d(x,z)$.

Proposition 5. *Let* $(\bar{x}, \bar{z})$ *satisfy* $\bar{c}(\bar{x}, \bar{z}) = 0$ *with* $g(\bar{x}) \leq 0$. *Let* G *be any compact subset of* $\Re^{n+m}$ *containing* $(\bar{x}, \bar{z})$ *and satisfying:*

i) **A1** *and* **A4** *hold for all* $(x,z) \in G$ *with* $\bar{c}(x,z) = 0$;

ii) there is no $(x,z) \in G$ *with* $\bar{c}(x,z) = 0$ *such that* x *is a solution to (1.1).*

Then for each $d > 0$ *there exits an* $\epsilon(d) > 0$ *such that for all* $(x,z) \in G$ *with* $r(x,z) < \epsilon(d)$,

$$\nabla \psi_d(x,z)^{\mathsf{T}} \begin{pmatrix} \delta \\ q \end{pmatrix} < 0.$$

Proof: Let $(x,z) \in G$ with $\bar{c}(x,z) = 0$. Then

$$\nabla \psi_d(x,z) = \begin{pmatrix} \nabla f(x) \\ 0 \end{pmatrix} + \begin{pmatrix} \nabla g(x) \\ I \end{pmatrix} \bar{\lambda}(x,z).$$

From (2.5) and (2.9), it follows that

$$\nabla \psi_d(x,z)^{\mathsf{T}} \begin{pmatrix} \delta \\ q \end{pmatrix} = -\delta^{\mathsf{T}} B \delta - \lambda \nabla g(x)^{\mathsf{T}} \delta$$

where λ is the multiplier vector for (1.2). But if λ_j is positive then it is the case that $(\nabla g(x)^{\mathsf{T}} \delta + g(x))_j = 0$ and, since $g_j(x) \leq 0$,

$$\lambda^{\mathsf{T}} \nabla g(x)^{\mathsf{T}} \delta \geq 0.$$

Thus

$$\nabla \psi_d(x,z)^{\mathsf{T}} \begin{pmatrix} \delta \\ q \end{pmatrix} \leq -\delta^{\mathsf{T}} B \delta < 0$$

unless $\delta = 0$ (which implies that x is a solution to (1.1)). The result now follows by a continuity argument.

□

The preceding proposition shows that the direction of the step in (x, z) is a descent direction for our merit function when $\bar{c}(x, z)$ is close to zero except possibly in a neighborhood of (x^*, z^*) where x^* is a solution to (1.1) and $z^* = -g(x^*)$. We now patch that gap by showing that if x^* is a strong solution then the step is a descent direction for $\psi_d(x, z)$. First we recall that if x is close enought to x^* then the active constraints of (1.2) at x are the same as the active constraints of (1.1) at x^*. Thus for x sufficiently close to x^*, (1.2) is equivalent to

$$\begin{array}{l} \min_\delta \nabla f(x)^\mathsf{T}\delta + \frac{1}{2}\delta^\mathsf{T} B\delta \\ \text{subject to: } \nabla g_a(x)^\mathsf{T}\delta + g_a(x) = 0 \end{array} \tag{2.11}$$

where (g_a, g_u) denotes the partition into active and inactive constraints at x^*. It follows from the theory for equality-constrained SQP methods (see, for example, [BogTW82]) that for x sufficiently close to x^* there is a positive constant K such that the solution of (2.11) satisfies

$$|\delta| \leq K|x - x^*|.$$

If, in addition, z is close to $z^* = -g(x^*)$ and q is given by (2.4) then using the partition above we obtain

$$q_a = -\left(\nabla g_a(x)^\mathsf{T}\delta + g_a(x) + z_a\right) = -z_a$$

so

$$|q_a| \leq |z_a| = |z_a - z_a^*|.$$

Since $z_u^* = -g_u(x^*)$,

$$\begin{array}{rcl} |q_u| & = & |\nabla g_u(x)^\mathsf{T}\delta + g_u(x) - g_u(x^*) - z_u^* + z_u| \\ & \leq & K_1|x - x^*| + K_2|x - x^*| + |z_u - z_u^*| \end{array}$$

for some constants K_1 and K_2. Thus δ and q are both of order $(|x - x^*| + |z - z^*|)$. We can now prove the following.

Proposition 6. *Let x^* be a strong solution to (1.1) with multiplier vector λ^*. Let $z^* = -g(x^*)$ and assume that* **A5** *holds. Then there exist positive constants η and ξ such that*

$$\nabla\psi_d(x, z)^\mathsf{T}\begin{pmatrix} \delta \\ q \end{pmatrix} < 0$$

whenever $0 < |(x, z) - (x^*, z^*)| \leq \eta$ and $0 < d \leq \xi$.

Proof: We assume η is sufficiently small that the bounds on δ and q preceding the statement of the proposition are valid. By the remarks following Proposition 1 it is seen that $\bar{A}(x,z)^{-1}$ is well-defined and uniformly positive definite for η small. From (2.5) – (2.8),

$$\nabla\psi_d(x,z)^{\mathsf{T}}\begin{pmatrix}\delta\\ q\end{pmatrix} = \nabla f(x)^{\mathsf{T}}\delta + \bar{\lambda}(x,z)^{\mathsf{T}}w - \frac{2}{d}w\bar{A}(x,z)^{-1}w + O(p|w|) + \frac{1}{d}O(p|w|^2)$$

where $p = \min\{|\delta|, |q|\}$ and $w = \left(\nabla g(x)^{\mathsf{T}}\delta + q\right)$. Letting λ be the multiplier vector for (1.2), using (2.9) and (2.10), and simplifying gives

$$\nabla\psi_d(x,z)^{\mathsf{T}}\begin{pmatrix}\delta\\ q\end{pmatrix} = -\delta^{\mathsf{T}}B\delta + q^{\mathsf{T}}\lambda - \frac{2}{d}w\bar{A}(x,z)^{-1}w + O(p|w|) + \frac{1}{d}O(p|w|^2).$$

Now $q_u^{\mathsf{T}}\lambda_u = 0$ and $q_a^{\mathsf{T}}\lambda_a = -z_a^{\mathsf{T}}\lambda_a \leq 0$, so using the uniform positive definiteness properties of the B and $\bar{A}(x,z)^{-1}$ we have that there are positive constants $\gamma_1, \gamma_2, \gamma_3$, and γ_4 such that

$$\nabla\psi_d(x,z)^{\mathsf{T}}\begin{pmatrix}\delta\\ q\end{pmatrix} \leq -\gamma_1 p^2 + \gamma_2 p|w| - \frac{1}{d}\gamma_3|w|^2 + \frac{1}{d}\gamma_4 p|w|^2.$$

It can be seen that for η and d sufficiently small that the right-hand side is negative and the proposition is proven.

□

A final important property of the merit function is that it doesn't impede the progress of superlinear convergence. That is, if the sequence is converging superlinearly then a steplength of one is acceptable near the solution. Since, as stated above, the quadratic program will identify the correct active constraints near the solution, the value of δ in that area will be given as the solution of (2.11). If we denote by P_a the projection onto the space orthogonal to the gradients of the active constraints at x^* then the characterization of the superlinear convergence of the sequence $\{x^k\}$ generated by the SQP algorithm is ([BogTW82]),

$$\lim_{k\to\infty}\frac{|P_a(\nabla^2_{xx}L(x^k,\lambda^k) - B^k)\delta^k|}{|\delta^k|} = 0 \qquad (2.12)$$

where (δ^k, λ^k) is the optimal solution-multiplier pair for the quadratic program (2.11).

Proposition 7. *Let the hypotheses of Proposition 6 hold. Assume that at least one step with steplength one is taken after the correct active set is identified for the sequence $\{x^k\}$ generated by the SQP and that $\{x^k\}$ converges to x^* superlinearly. Then there exists a $\bar{d} > 0$ such that for each $d \in (0, \bar{d})$ there is a positive integer $J(d)$ satisfying*

$$\psi_d(x^k + \delta^k, z^k + q^k) < \psi_d(x^k, z^k)$$

for $k \geq J(d)$.

Proof: For ease of notation we drop the superscript k henceforth. Defining p and w as in the proof of Proposition 6 and using equations (2.5) – (2.10) it follows from the Taylor Series that

$$\begin{aligned} \Delta &= \psi_d(x+\delta, z+q) - \psi_d(x,z) \\ &= -\delta^T B\delta + \frac{1}{2}\delta^T \nabla_{xx} L(x,\lambda)\delta + \lambda^T q \\ &\quad -q^T \Lambda \bar{A}(x,z)^{-1} w + \delta^T B \nabla g(x) \bar{A}(x,z)^{-1} w \\ &\quad -\frac{1}{d} w^T \bar{A}(x,z)^{-1} w \ + (1 + \frac{1}{d}) O(p^3). \end{aligned} \tag{2.13}$$

If x is close enough to the solution that the active set has been identified at each iterate $(\nabla g_a(x)^T \delta + g_a = 0)$ then

$$z_a + q_a = -(\nabla g_a(x)^T \delta + g_a) = 0$$

and hence after one step of unit length $z_a = q_a = 0$. Since $\lambda_u = 0$ we conclude that $\lambda^T q = 0$ and $\Lambda q = 0$. Now letting $Q_a = I - P_a$ and adding and subtracting $\frac{1}{2}\delta^T B\delta$ on the right-hand side of (2.13) we obtain

$$\begin{aligned} \Delta &= -\frac{1}{2}\delta B\delta + \frac{1}{2}\delta^T P_a(\nabla_{xx} L(x,\lambda) - B)\delta \\ &\quad +\frac{1}{2}\delta^T Q_a(\nabla_{xx} L(x,\lambda) - B)\delta \\ &\quad -\frac{1}{d} w^T \bar{A}(x,z)^{-1} w + \delta^T B \nabla g(x) \bar{A}(x,z)^{-1} w \\ &\quad +(1 + \frac{1}{d}) O(p^3). \end{aligned} \tag{2.14}$$

Since $q_a = 0$, $\nabla g_a(x)^T \delta = w_a$ and hence

$$Q_a \delta = \nabla g_a(x) \left(\nabla g_a(x)^T \nabla g_a(x) \right)^{-1} \nabla g_a(x)^T \delta = O(w_a).$$

Thus letting $\epsilon = |w|/|\delta|$ in (2.14) and using **A5**, we can write

$$\begin{aligned} \frac{\Delta}{|\delta|^2} \leq & -\frac{1}{2}\mu + \bar{K}\epsilon - \frac{1}{d}\gamma\epsilon^2 \\ & + \frac{1}{2}\frac{|P_a(\nabla_{xx} L(x,\lambda) - B)\delta|}{|\delta|} + (1 + \frac{1}{d})O(p). \end{aligned}$$

From (2.12) it can be seen that for d sufficiently small the right-hand side can be made negative provided p is small, thus proving the result.

□

3. An Algorithm

From the basic theorems presented in the preceding section a global convergence theory can be constructed using the merit function $\psi_d(x, z)$. This theoretical development will not be pursued here as it does not differ significantly from that presented for the equality-constrained case considered in [BogT89] (§3). From a practical point of view the implementation of the algorithm is more important. Below we present details of our implementation along with results of some numerical experimentation.

Because the merit function $\psi_d(x, z)$ is relatively difficult to evaluate at points other than iterates and such evaluations are necessary when carrying out a line search, a modified version of the merit function is used in our implementation. We define our approximate merit function at the kth iteration to be

$$\psi_d^k(x, z) = f(x) + (g(x) + z)^T \bar{\lambda}^k + \frac{1}{d}(g(x) + z)^T \bar{A}_k^{-1}(g(x) + z)$$

where

$$\bar{A}_k = \nabla g(x^k)^T \nabla g(x^k) + Z^k$$

and

$$\bar{\lambda}^k = -\bar{A}_k^{-1} \nabla g(x^k)^T \nabla f(x^k).$$

Note that the use of $\psi_d^k(x, z)$ as a merit function requires only the evaluation of f and g to test a prospective new iterate. While it is cheap to evaluate,

however, it is not a single merit function, and thus needs to be combined with a strategy that will ensure global convergence.

The relevant properties of $\psi_d^k(x,z)$ are identical to those of $\psi_d(x,z)$ provided in the preceding section except that the descent property of $\psi_d^k(x,z)$ is stronger than that for $\psi_d(x,z)$. In fact, (δ^k, q^k) is *always* a descent direction for $\psi_d^k(x,z)$, for d sufficiently small, as is a consequence of the following proposition.

Proposition 8. *Let C be a compact subset of $\Re^n$ not containing a solution of (1.1) and in which $\bar{A}_k$ is nonsingular. There exists a $\bar{d}$ sufficiently small such that if (x^k, z^k) is a point of C at which $\nabla\psi_d^k(x^k,z^k) \neq 0$ then*

$$\nabla\psi_d^k(x^k,z^k)^{\mathsf{T}} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix} < 0$$

provided $d \leq \bar{d}$.

Proof: It is seen that

$$\begin{aligned}
\Delta &= \nabla\psi_d^k(x^k,z^k)^{\mathsf{T}} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix} \\
&= \nabla f(x^k)^{\mathsf{T}}\delta^k \\
&+ (\bar{\lambda}^k)^{\mathsf{T}} \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix} \\
&- \frac{1}{d} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix} \bar{A}_k^{-1} \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix}.
\end{aligned}$$

Using (2.9) and (2.10) yields, after some cancellation,

$$\begin{aligned}
\Delta &= -(\delta^k)^{\mathsf{T}} B^k \delta^k + \lambda^{\mathsf{T}} q^k \\
&+ (\delta^k)^{\mathsf{T}} B^k \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix} \bar{A}_k^{-1} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix} - (q^k)^{\mathsf{T}} \Lambda \bar{A}_k^{-1} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix} \\
&- \frac{1}{d} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix} \bar{A}_k^{-1} \begin{pmatrix} \nabla g(x^k) \\ I \end{pmatrix}^{\mathsf{T}} \begin{pmatrix} \delta^k \\ q^k \end{pmatrix}.
\end{aligned}$$

As in the proof of Proposition 6, $\lambda^{\mathsf{T}} q^k \leq 0$. Defining w as in the same proof it follows that there exist a constant K such that

$$\Delta \leq -(\delta^k)^{\mathsf{T}} B^k \delta^k + K|w| - \frac{1}{d}|w|\bar{A}_k^{-1}|w|.$$

Using assumption **A5** it is seen that the right hand side of the above inequality has a negative maximum for all d sufficiently small.

□

The proof for the case when x^k is in a neighborhood of a solution to (1.1) is nearly identical to that of Proposition 6 and is not given here. Similarly the proof that if the algorithm is converging q-superlinearly, then a steplength of one is acceptable for $\psi_d^k(x,z)$ near the solution parallels the proof of Proposition 7.

Our strategy for global convergence, as developed in [BogT89], depends on the fact that there is a positive constant η such that if $|g(x)+z| \leq \eta$ then both the function $\psi_d(x,z)$ and $\psi_d^k(x,z)$ are reduced in the direction (δ, q). For a given η, if $|g(x)+z| \leq \eta$ then our strategy is to insist that $\psi_d^k(x,z)$ be reduced. In most cases, $\psi_d(x,z)$ will then be reduced also. If not, then the current value of η is too large and it is reduced. Generally, we prefer η not to be too small initially so that the constraints are not followed too closely. If $|g(x)+z| > \eta$ our procedures call for both $\psi_d^k(x,z)$ and $|g(x)+z|$ to be reduced. This is possible because of the stronger descent property of $\psi_d^k(x,z)$. The details are given in the algorithm description below. This description assumes that we have an appropriate value of d; we comment on this problem later.

ALGORITHM

1. Given $d > 0$, x^0, B^0:
   Calculate $f(x^0)$, $\nabla f(x^0)$, $g(x^0)$, $\nabla g(x^0)$ and $c(x^0, z^0)$.
   Initialize ${z^0}_i = -g_i$ for all $g_i < 0$ and ${z^0}_i = 1$ otherwise, and set $\eta = 2\left|g(x^0)\right|^2$.
   Calculate $\bar{\lambda}^0$ and $\psi_d(x^0, z^0)$, and set $k = 0$.

2. Solve

$$\min_{\delta} \nabla f(x^k)^{\mathsf{T}}\delta + \tfrac{1}{2}\delta^{\mathsf{T}} B^k \delta$$

$$\text{subject to: } \nabla g(x^k)^{\mathsf{T}}\delta + g(x^k) \leq 0$$

   to obtain δ^k.

3. Set $q^k = -[\nabla g(x^k)^{\mathsf{T}}\delta^k + g(x^k) + z^k]$.

4. Compute $\alpha^k \leq 1$ such that

$$\psi_d^k(x^k + \alpha^k\delta^k, z^k + \alpha^k q^k) < \psi_d^k(x^k, z^k).$$

   (If this fails numerically, then choose $\alpha^k \leq 1$ to reduce $|c(x,z)|$.)

5. If $\left|c(x^k + \alpha^k\delta^k, z^k + \alpha^k q^k)\right| \geq \left|c(x^k, z^k)\right|$ and $\left|c(x^k, z^k)\right| > \eta$, reduce α^k until $\left|c(x^k + \alpha^k\delta^k, z^k + \alpha^k q^k)\right| < \left|c(x^k, z^k)\right|$.
6. If $\psi_d(x^k + \alpha^k\delta^k, z^k + \alpha^k q^k) > \psi_d(x^k, z^k)$ set $\eta = \frac{1}{2}\left|c(x^k, z^k)\right|$.
7. Set

$$\begin{aligned} x^{k+1} &:= x^k + \alpha^k\delta^k \\ z^{k+1} &:= z^k + \alpha^k q^k. \end{aligned}$$

8. Update B^k.
9. If convergence criteria are met, stop.
10. Set $k := k + 1$; goto 2.

The algorithm described in this paper was implemented in Fortran and tested on a selection of problems from [HocS81] and [Sch87]. The problems chosen were the higher dimensional ones with a large number of nonlinear inequality constraints. (The low dimensional problems converged too quickly to be of much interest.) Because equality constraints can be handled in a straightforward manner [BogT89], no problems with equality constraints were chosen. A variety of starting points were used, the first of which was the one suggested in the reference. The tests were run on a Sparc 3333 station in double precision arithmetic (16 decimal digits). The quadratic subproblem was solved using the Nag subroutine E04NAF. We found it important to update the approximation of the Hessian of the Lagrangian at each iteration, while still maintaining positive definiteness. Accordingly we used the numerically proven damped BFGS update of [Pow78], defined as follows. In the standard BFGS one uses the vectors

$$\begin{aligned} s &= x^{k+1} - x^k \\ y &= \nabla_x L(x^{k+1}, \lambda^{k+1}) - \nabla_x L(x^k, \lambda^{k+1}) \end{aligned}$$

in the update formula. Since $s^{\mathsf{T}}y$ need not be positive, Powell recommends replacing y by

$$y^P = \theta y + (1 - \theta)B_k s$$

where the parameter $\theta \in (0, 1]$ is chosen as

$$\theta = \begin{cases} 1 & \text{if } s^T y \geq \zeta s^T B^k s \\ (1 - \zeta)s^{\mathsf{T}}B_k s/(s^{\mathsf{T}}B_k s - s^T y) & \text{otherwise} \end{cases}$$

and the value of ζ was chosen to be .1. More theoretically justified updating schemes exist, e.g., [Tap88], but because of their local nature, they were not employed here.

A pre-update scaling strategy suggested for unconstrained optimization [OreS76] was used. This pre-update scaling was only performed on the initial iteration as suggested by [ShaP78]. B^0 is set to I and the first step is computed; s and y are calculated. Before the above updating procedure is used, however, B^0 is set to τI where

$$\tau = \begin{cases} s^T y / s^T s & \text{if } s^T y > 0 \\ 1 & \text{otherwise} \end{cases}.$$

A dynamic scheme for decreasing the parameter d was devised and employed in [BogT89]. It was noticed in testing that the parameter was decreased infrequently and that an initial fixed value of d would, in most cases, yield adequate convergence. A procedure that decreases d frivolously can be very costly, as it will usually lead to smaller steps. For these reasons, the updating of the parameter d was not included in the algorithm tested here; rather, several different values of the parameter were tested for each problem. (See Remark 5 below.)

The convergence criteria used are those that are standard in the literature. We first insist that the constraints be satisfied to a close tolerance; specifically we require

$$\left| g_v(x^k) \right| \leq 10^{-7}$$

where $v \in \{i \ : \ g_i(x^k) > 0\}$. We also require that either

$$\frac{\left| \nabla f(x^k) + \nabla g(x^k) \lambda^k \right|}{|f(x^k)|} \leq 10^{-7} \tag{3.1}$$

or

$$\left| x^k - x^{k-1} \right| \leq 10^{-8}(1 + \left| x^k \right|). \tag{3.2}$$

As noted above, the test problems are taken from [HocS81] and [Sch83]. In Table 1 we give the problem number from these references, some relevant information about the problem, and the performance of the algorithm on that problem. In particular, we show the dimension, n, of the problem; the number of constraints, m; the number of those constraints that are simple bounds, b; and the number of active constraints at the solution. For each problem we consider several starting values and several values of the parameter d. Then for each of these runs we report the number of

iterations (which is equal to the number of evaluations of $\nabla f(x)$ and $\nabla g(x)$); the number of function and constraint evaluations; the iteration, $iter_{act}$, at which the correct active set was identified; the number of iterations where α^k was equal to 1; the number of times that η had to be reduced (see Step 6); the number of times that $\psi_d^k(x,z)$ could not be reduced (see Step 4); and the convergence criteria satisfied where 1 implies that test (3.1) was satisfied, 2 implies that (3.2) was satisfied, 3 implies that both were, 7 implies numerical overflow in a function evaluation, and 8 signifies that an inconsistent QP was detected.

We make several observations concerning these results.

1. No attempt was made to optimize the code. For example, simple bounds should be handled separately and not be included in the merit function since they will always be satisfied. In the same spirit, the gradients of linear constraints should only be computed once. Finally, a more sophisticated line search should be used, and a procedure for reducing constraint infeasibilities when the QP is inconsistent should be included.

2. In light of the above remark, we did not attempt a detailed comparison with other methods. Direct comparisons are confounded by the host of other factors such as the updating strategy, line search, and differences in the machine on which the codes were run. We note, however, that our results on the "standard" starting values are similar to those of [HocS81] and [Sch83] even though they did not use the same stopping criteria.

3. The merit function performs well. Most of the steps have $\alpha^k = 1$, so no other merit function would do substantially better. The worst performance occurred in the examples where there was a significant amount of linear dependency in the active constraint gradients (i.e., half or more of the gradients are dependent). We think that these problems, however, are unrealistic, especially in the large scale case. In problems like No. 101 where only 9 of 64 steps have $\alpha^k = 1$, we observed that almost all of these steps occurred after the correct active set was identified.

4. Our procedure for η caused no apparent problems; the code performed as in the equality constrained case reported in [BogT89].

5. The algorithm does not appear to be overly sensitive to the "arbitrary" parameter d. Occasionally smaller values of d help, sometimes not, and often there is little difference. We think that this is due to

the fact that most of the problems are well scaled. To test this, we multiplied the objective function by 1000. Occasionally this did cause some significant differences. Moreover, we note that there are several instances where the line search in Step 4 failed and the procedure for reducing $|\bar{c}(x,z)|$ was invoked. While this procedure was successful, we think that this and the above observation suggest the need to dynamically reduce d to ensure that (δ, q) is a good descent direction for ψ_d^k.

In conclusion, we believe that the merit function and the algorithm described here are competitive procedures for solving general nonlinear programming problems.

REFERENCES

[BogT84] Boggs, P., Tolle, J., *A family of descent functions for constrained optimization,* SIAM J. Numer. Anal. 21 (1984), pp. 1146-1161.

[BogT88] Boggs, P., Tolle, J., *Merit functions and nonlinear programming,* in Operational Research '87, G. K. Rand, ed., Elsevier Science Publishers B.V., North Holland, 1988.

[BogT89] Boggs, P., Tolle, J., *A strategy for global convergence in a sequential quadratic programming algorithm,* SIAM J. Numer. Anal. 26 (1989), pp. 600-623.

[BogTW82] Boggs, P., Tolle, J., Wang, P., *On the local convergence of quasi-Newton methods for constrained optimization,* SIAM J. Control Optim. 20 (1982), pp. 161-171.

[ChaLLP82] Chamberlain, R., Lemarechal, C., Pedersen, H. C., and Powell, M., *The watchdog technique for forcing convergence in algorithms for constrained optimization,* Math. Programming Stud. 16 (1982), pp. 1-17.

[GilMSW86] Gill, P., Murray, W., Saunders, M., and Wright, M., *Some theoretical properties of an augmented Lagrangian merit function,* Technical Report 86-6, Stanford University, Dept. Operations Research, 1986.

[Han77] Han, S., *A globally convergent method for nonlinear programming,* J. Optim. Theory Appl. 22 (1977), pp. 297-309.

[HocS81] Hock, W., Schittkowski. K., *Test examples for nonlinear programming codes*, in Lecture Notes in Economics and Mathematical Systems 187, Springer-Verlag, Berlin, 1981.

[OreS76] Oren, S., Spedicato, E.,*Optimal conditioning of self-scaling variable metric algorithms,* Math. Programming. 10 (1976), pp. 70-90.

[Pow78] Powell, M. J. D., *A fast algorithm for nonlinearly constrained optimization calculation.* in Numerical Analysis Proceedings Dundee 1977, G. A. Watson, editor, Springer-Verlag, 1978.

[Pow84] Powell, M. J. D., *The performance of two subroutines for constrained optimization on some difficult test problems,* in Numerical Optimization 1984, P. T. Boggs, R. H. Byrd, and R. Schnabel, editors. SIAM, 1984.

[Rob74] Robinson, S., *Perturbed Kuhn-Tucker points and rates of convergence for a class of nonlinear-programming algorithms,* Math. Programming 7 (1974), pp. 1-16.

[Sch83] Schittkowski, K., *On the convergence of a sequential quadratic programming method with an augmented Lagrangian line search function,* Math. Operationsforsch. Statist. Ser. Optim. 14 (1983), pp. 197-216.

[Sch87] Schittkowski, K., *More test examples for nonlinear programming codes*, in Lecture Notes in Economics and Mathematical Systems 282, Springer-Verlag, Berlin, 1987.

[ShaP78] Shanno, D., Phua, K., *Matrix conditioning and nonlinear optimization*, Math. Programming 14 (1978), pp. 145-160.

[ShaP89] Shanno, D., and Phua, K., *Numerical experience with sequential quadratic programming algorithms for equality constrained nonlinear programming*, acm Trans. Math. Soft. 15 (1989), pp. 49 - 63.

[Tap77] Tapia, R., *Diagonalized multiplier methods and quasi-Newton methods for constrained optimization*, J. Optim. Theory Appl. 22 (1977), pp. 135-194.

[Tap88] Tapia, R., *On secant updates for use in general constrained optimization*, Math. Comp. 51 (1988), pp. 181-202.

TABLE 1

Problem 72, $n = 4$, $m = 2$, $b = 8$; 2 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	17	141	141	1	2	7	5
	0.1	17	130	130	1	2	7	4
	0.01	17	120	120	1	2	7	3
	0.001	17	120	120	1	2	7	3
Medium	1.0	16	135	135	1	2	6	5
	0.1	16	115	115	1	2	6	3
	0.01	16	115	115	1	2	6	3
	0.001	16	115	115	1	2	6	3
Far	1.0	19	195	195	14	2	11	9
	0.1	20	175	175	13	2	11	7
	0.01	20	165	165	13	2	11	6
	0.001	20	165	165	13	2	11	6
$f \times 1000$	1.0	25	290	290	19	2	15	16
	0.1	25	290	290	19	2	15	16
	0.01	25	290	290	19	2	14	16
	0.001	18	158	290	2	2	8	6

Problem 84, $n = 5$, $m = 6$, $b = 10$; 5 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	2	28	28	1	1	0	1
	0.1	2	28	28	1	1	0	1
	0.01	2	28	28	1	1	0	1
	0.001	2	28	28	1	1	0	1
Medium	1.0	4	40	40	2	3	0	1
	0.1	4	30	30	2	3	0	0
	0.01	4	30	30	2	3	0	0
	0.001	4	31	31	2	3	1	0
Far	1.0	3	24	24	3	1	0	0
	0.1	3	24	24	3	1	0	0
	0.01	4	51	51	2	2	0	2
	0.001	4	41	41	2	2	0	1
$f \times 1000$	1.0	2	28	28	1	1	0	1
	0.1	2	28	28	1	1	0	1
	0.01	2	28	28	1	1	0	1
	0.001	2	28	28	1	1	0	1

Problem 93, $n = 6$, $m = 2$, $b = 6$; 2 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	24	188	188	1	1	0	1
	0.1	24	188	188	1	1	1	1
	0.01	20	151	151	1	2	1	0
	0.001	18	138	138	1	2	0	0
Medium	1.0	20	159	159	4	2	2	1
	0.1	20	159	159	4	2	1	1
	0.01	23	171	171	4	2	0	0
	0.001	23	171	171	4	2	0	0
Far	1.0	34	406	406	31	2	17	16
	0.1	26	231	231	2	1	3	4
	0.01	26	202	202	2	1	0	1
	0.001	24	179	179	2	1	0	0
$f \times 1000$	1.0	> 100				8		
	0.1	> 100				8		
	0.01	> 100				8		
	0.001	> 100				8		

Problem 100, $n = 7$, $m = 4$, $b = 0$; 2 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	19	196	197	2	1	3	3
	0.1	19	196	197	2	1	3	3
	0.01	10	93	93	2	1	0	0
	0.001	12	112	112	3	1	0	0
Medium	1.0	41	365	365	20	2	4	2
	0.1	41	374	375	20	2	2	3
	0.01	41	364	365	20	2	3	2
	0.001	42	386	388	20	1	4	2
Far	1.0	24	236	236	6	1	3	3
	0.1	26	255	255	9	1	2	3
	0.01	26	245	245	9	1	4	2
	0.001	31	294	294	12	1	4	3
$f \times 1000$	1.0					7		
	0.1					7		
	0.01					7		
	0.001					7		

Problem 101, $n = 7$, $m = 22$, $b = 16$; 3 active constraints*

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	33	458	507	30	3	17	18
	0.1	63	916	1023	45	1	35	31
	0.01	64	793	907	44	1	19	19
	0.001	64	784	907	44	1	14	19
Medium	1.0					7		
	0.1					7		
	0.01					7		
	0.001					7		
Far	1.0					7		
	0.1					7		
	0.01					7		
	0.001					7		
$f \times 1000$	1.0					7		
	0.1					7		
	0.01					7		
	0.001					7		

Problem 104, $n = 8$, $m = 6$, $b = 16$; 4 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	25	234	234	6	1	0	0
	0.1	25	234	234	6	1	1	0
	0.01	27	261	261	6	1	1	0
	0.001	33	362	362	6	1	3	3
Medium	1.0	16	179	167	5	1	2	0
	0.1	17	181	171	5	1	3	0
	0.01	17	183	171	5	1	2	0
	0.001	18	195	181	5	1	2	0
Far	1.0	24	238	238	9	1	0	1
	0.1	27	259	259	9	1	0	0
	0.01	33	336	338	9	1	3	1
	0.001	40	429	429	9	1	2	1
$f \times 1000$	1.0					7		
	0.1					7		
	0.01					7		
	0.001					7		

Problem 106, $n = 8$, $m = 6$, $b = 16$; 6 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	23	270	294	20	2	3	4
	0.1	31	373	415	27	2	7	6
	0.01	32	382	424	27	2	7	6
	0.001	58	830	1038	54	2	29	27
Medium	1.0	12	121	121	8	2	1	0
	0.1	10	112	112	7	2	3	1
	0.01	10	112	112	7	2	3	1
	0.001	10	112	112	7	2	3	1
Far	1.0	23	321	370	21	2	10	9
	0.1	35	473	598	34	2	13	13
	0.01	32	435	542	31	2	12	13
	0.001	26	362	440	24	2	10	11
$f \times 1000$	1.0	21	278	311	16	2	8	8
	0.1	19	240	251	12	1	4	6
	0.01	15	169	169	8	1	1	2
	0.001	14	157	157	8	1	3	1

Problem 108, $n = 9$, $m = 13$, $b = 1$; 9 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	10	150	161	8	1	1	4
	0.1	10	120	131	8	1	1	1
	0.01	10	120	131	8	1	1	1
	0.001	10	120	131	8	1	1	1
Medium	1.0	18	205	205	17	1	0	1
	0.1	18	214	214	16	1	0	1
	0.01	22	329	365	26	2	5	3
	0.001	33	434	491	32	2	5	4
Far	1.0	16	176	176	16	1	0	0
	0.1	16	176	176	16	1	0	0
	0.01	16	187	187	16	1	0	1
	0.001	16	177	177	16	1	0	0
$f \times 1000$	1.0	10	138	139	5	1	1	2
	0.1	10	138	139	5	1	1	2
	0.01	10	129	129	5	1	1	1
	0.001	10	138	139	5	1	1	2

Problem 110, $n = 10$, $m = 0$, $b = 20$; 0 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	7	89	89	2	1	0	0
	0.1	7	89	89	2	1	0	0
	0.01	7	89	89	2	1	0	0
	0.001	7	89	89	2	1	0	0
Medium	1.0	9	115	115	1	1	0	0
	0.1	9	115	115	1	1	0	0
	0.01	9	115	115	1	1	0	0
	0.001	9	115	115	1	1	0	0
Far	1.0	16	189	189	2	1	0	0
	0.1	16	189	189	2	1	0	0
	0.01	16	189	189	2	1	0	0
	0.001	16	189	189	2	1	0	0
$f \times 1000$	1.0	7	89	89	2	1	0	0
	0.1	7	89	89	2	1	0	0
	0.01	7	89	89	2	1	0	0
	0.001	7	89	89	2	1	0	0

Problem 113, $n = 10$, $m = 8$, $b = 0$; 6 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	18	211	211	8	1	0	0
	0.1	18	212	212	7	1	0	0
	0.01	18	213	213	7	1	0	0
	0.001	18	215	215	7	1	1	0
Medium	1.0	14	167	167	8	1	1	0
	0.1	14	168	168	8	1	1	0
	0.01	16	196	196	9	1	2	0
	0.001	16	196	196	9	1	2	0
Far	1.0	17	221	221	11	1	0	2
	0.1	17	202	202	11	1	1	0
	0.01	17	221	221	11	1	3	2
	0.001	17	232	232	10	1	3	3
$f \times 1000$	1.0	25	367	465	21	2	5	8
	0.1	25	367	465	21	2	5	8
	0.01	25	367	465	21	2	5	8
	0.001	25	367	465	21	2	5	8

Problem 264, $n = 4$, $m = 3$, $b = 0$; 1 active constraint

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	11	105	105	2	1	2	3
	0.1	12	112	112	2	2	2	3
	0.01	17	148	181	14	2	2	4
	0.001	15	124	138	15	2	2	2
Medium	1.0	11	91	91	2	1	5	3
	0.1	12	97	97	3	1	2	3
	0.01	11	63	63	3	1	1	0
	0.001	12	67	67	3	1	1	0
Far	1.0	12	106	106	5	1	3	4
	0.1	12	88	88	4	1	1	2
	0.01	12	67	67	4	1	1	0
	0.001	12	68	68	4	1	2	0
$f \times 1000$	1.0	17	166	177	1	2	5	7
	0.1	17	166	177	1	2	5	7
	0.01	17	166	177	1	2	5	7
	0.001	17	166	177	1	2	5	7

Problem 339, $n = 3$, $m = 1$, $b = 3$; 1 active constraint

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	11	105	105	2	1	2	3
	0.1	12	112	112	2	2	2	3
	0.01	17	148	181	14	2	2	4
	0.001	15	124	138	15	2	2	2
Medium	1.0	11	91	91	2	1	5	3
	0.1	12	97	97	3	1	2	3
	0.01	11	63	63	3	1	1	0
	0.001	12	67	67	3	1	1	0
Far	1.0	12	106	106	5	1	3	4
	0.1	12	88	88	4	1	1	2
	0.01	12	67	67	4	1	1	0
	0.001	12	68	68	4	1	2	0
$f \times 1000$	1.0	17	166	177	1	2	5	7
	0.1	17	166	177	1	2	5	7
	0.01	17	166	177	1	2	5	7
	0.001	17	166	177	1	2	5	7

Problem 346, $n = 3$, $m = 2$, $b = 6$; 1 active constraint

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	3	16	16	1	1	0	1
	0.1	3	16	16	1	1	1	0
	0.01	3	16	16	1	1	1	0
	0.001	3	16	16	1	1	1	0
Medium	1.0	7	69	69	6	1	1	3
	0.1	12	114	120	10	3	7	6
	0.01	12	114	120	10	3	7	5
	0.001	12	114	120	10	3	7	3
Far	1.0	6	48	48	2	1	1	2
	0.1	6	48	48	2	1	1	2
	0.01	6	48	48	2	1	1	2
	0.001	6	48	48	2	1	1	2
$f \times 1000$	1.0	3	26	26	1	1	2	1
	0.1	3	26	26	1	1	2	1
	0.01	3	26	26	1	1	1	1
	0.001	3	26	26	1	1	0	1

Problem 354, $n = 4$, $m = 1$, $b = 4$; 1 active constraint

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x,z)$ not reduced
Standard	1.0	24	130	130	12	2	0	0
	0.1	24	130	130	12	2	0	0
	0.01	24	130	130	12	2	0	0
	0.001	24	130	130	12	2	0	0
Medium	1.0	48	248	248	32	3	0	0
	0.1	48	248	248	32	3	0	0
	0.01	48	248	248	32	3	0	0
	0.001	48	248	248	32	3	0	0
Far	1.0	39	203	203	27	3	0	0
	0.1	39	203	203	27	3	0	0
	0.01	39	203	203	27	3	0	0
	0.001	39	203	203	27	3	0	0
$f \times 1000$	1.0	24	130	130	12	2	0	0
	0.1	24	130	130	12	2	0	0
	0.01	24	130	130	12	2	0	0
	0.001	24	130	130	12	2	0	0

Problem 387, $n = 4$, $m = 3$, $b = 0$; 10 active constraints

Starting value	d	iter	No. f	No. g	$iter_{act}$	conv.	No. η reductions	No. $\psi_d^k(x, z)$ not reduced
Standard	1.0	54	1140	1152	47	2	19	15
	0.1	51	1131	1189	50	2	19	21
	0.01	51	1131	1188	50	2	19	20
	0.001	51	1131	1191	50	2	19	19
Medium	1.0	25	520	520	17	1	11	10
	0.1	25	520	520	17	1	8	10
	0.01	27	511	512	17	1	7	6
	0.001	26	457	457	18	1	2	2
Far	1.0	27	569	570	19	1	13	12
	0.1	26	543	544	19	1	11	11
	0.01	26	504	506	19	1	5	7
	0.001	28	518	518	19	1	3	5
$f \times 1000$	1.0	23	494	494	2	13	8	9
	0.1	23	494	494	2	13	8	9
	0.01	23	494	494	2	13	8	9
	0.001	23	494	494	2	13	8	9

NIST-114A
(REV. 3-90)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY

BIBLIOGRAPHIC DATA SHEET

1. PUBLICATION OR REPORT NUMBER
NISTIR 4702

2. PERFORMING ORGANIZATION REPORT NUMBER

3. PUBLICATION DATE
October 1991

4. TITLE AND SUBTITLE

A Merit Function for Inequality Constrained Nonlinear Programming Problems

5. AUTHOR(S)

Paul T. Boggs and Jon W. Tolle

6. PERFORMING ORGANIZATION (IF JOINT OR OTHER THAN NIST, SEE INSTRUCTIONS)

U.S. DEPARTMENT OF COMMERCE
NATIONAL INSTITUTE OF STANDARDS AND TECHNOLOGY
GAITHERSBURG, MD 20899

7. CONTRACT/GRANT NUMBER

8. TYPE OF REPORT AND PERIOD COVERED

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS (STREET, CITY, STATE, ZIP)

10. SUPPLEMENTARY NOTES

11. ABSTRACT (A 200-WORD OR LESS FACTUAL SUMMARY OF MOST SIGNIFICANT INFORMATION. IF DOCUMENT INCLUDES A SIGNIFICANT BIBLIOGRAPHY OR LITERATURE SURVEY, MENTION IT HERE.)

We consider the use of the sequential quadratic programming (SQP) technique for solving the inequality constrained minimization problem $\min_x f(x)$ subject to: $g_i(x) \geq 0$, $i = 1, \ldots, m$. SQP methods require the use of an auxiliary function, called a merit function or line-search function, for assessing the steps that are generated. We derive a merit function by adding slack variables to create an equality constrained problem and then using the merit function developed earlier by the authors for the equality constrained case. We stress that we do not solve the slack variable problem, but only use it to construct the merit function. The resulting function is simplified in a certain way that leads to an effective procedure for updating the squares of the slack variables. The final form of the merit function has many desirable properties, and is demonstrated to be effective in practice.

12. KEY WORDS (6 TO 12 ENTRIES; ALPHABETICAL ORDER; CAPITALIZE ONLY PROPER NAMES; AND SEPARATE KEY WORDS BY SEMICOLONS)

nonlinear optimization; inequality constraints; sequential quadratic programming; merit functions

13. AVAILABILITY

[X] UNLIMITED

[ ] FOR OFFICIAL DISTRIBUTION. DO NOT RELEASE TO NATIONAL TECHNICAL INFORMATION SERVICE (NTIS).

[ ] ORDER FROM SUPERINTENDENT OF DOCUMENTS, U.S. GOVERNMENT PRINTING OFFICE, WASHINGTON, DC 20402.

[X] ORDER FROM NATIONAL TECHNICAL INFORMATION SERVICE (NTIS), SPRINGFIELD, VA 22161.

14. NUMBER OF PRINTED PAGES
33

15. PRICE
A03

ELECTRONIC FORM